

REMARKS

This is a full and timely response to the outstanding non-final Office Action mailed January 4, 2007. Claims 1-28 remain pending in the present application. Reconsideration and allowance of the application and pending claims are respectfully requested.

1. Response to Objections of Claims

Claims 5-6, 8, 12-14, 22, and 27 have been objected to under 37 CFR 1.75(c) as allegedly being in improper form because of multiple dependent claim issues. Claims 5-6, 8, 12-14, 22, and 27 have been amended to depend from single claims. Therefore, withdrawal of the objections is respectfully requested.

2. Response To Rejections of Claims Under 35 U.S.C. § 101

Claim 26 has been rejected under 35 U.S.C. § 101 as allegedly being directed to non-statutory subject matter. In particular, the Office Action objected to use of “means for” language in the claim. Accordingly, the claim has been amended to remove the “means for” language, and withdrawal of the rejection is respectfully requested.

3. Response To Rejections of Claims Under 35 U.S.C. § 102

Claims 1-4, 16-17, 19-21, and 23-26 have been rejected under 35 U.S.C. § 102(e) as allegedly being anticipated by *Burton* (U.S. Patent Publication No. 2003/0115379). Applicant respectfully traverses this rejection.

It is axiomatic that “[a]nticipation requires the disclosure in a single prior art reference of each element of the claim under consideration.” *W. L. Gore & Associates, Inc. v. Garlock, Inc.*, 721 F.2d 1540, 1554, 220 USPQ 303, 313 (Fed. Cir. 1983). Therefore, every claimed feature of the claimed subject matter must be represented in the applied reference to constitute a proper rejection under 35 U.S.C. § 102(e). In the present case, not every feature of the claimed subject matter is represented in the *Burton* reference. Applicant discusses the *Burton* reference and Applicant's claims in the following.

a. **Claim 1**

Per claim 1, Applicant claims:

A remote object invocation method for invoking a method of a remote object; the method comprising the steps of:

producing remote object data associated with the remote object to discover an object interface dynamically;

interpretatively establishing a proxy object using the remote object data at runtime of client software, the proxy object bearing an associated proxy method corresponding to the method of remote object;

invoking, in response to an action of the client software, the proxy object method;

conveying invocation data associated with the invocation of the proxy method to the remote object;

invoking, in response to the invocation data, the method of the remote object; and

returning invocation result data to the client software via the proxy object.

(Emphasis added).

Applicant respectfully submits that independent claim 1 is allowable for at least the reason that *Burton* does not disclose, teach, or suggest at least “producing remote object data associated with the remote object to discover an object interface dynamically; interpretatively establishing a proxy object using the remote object data at runtime of client software, the proxy object bearing an associated proxy method corresponding to the method of remote object; [and] invoking, in response to an action of the client software, the proxy object method,” as recited and emphasized above in claim 1.

Rather, *Burton* describes a system whereby a remote object and a proxy object are instantiated from a preexisting class. *Burton* describes that a client has a preexisting class (e.g., before runtime) for every object which it will ever instantiate. For example, *Burton* states that the “schema further includes proxy object subclasses 54a, 54b, and 54c that implement the proxy class for each of the remote classes 52a, 52b, and 52c. Accordingly, *Burton* fails to teach or suggest “producing remote object data associated with the remote object to discover an object interface dynamically; interpretatively establishing a proxy object using the remote object data at runtime of client software, the proxy object bearing an associated proxy method corresponding to the method of remote object; [and] invoking, in response to an

action of the client software, the proxy object method,” as recited and emphasized above in claim 1. (Emphasis added).

Thus, *Burton* fails to teach or suggest all of the features of claim 1. As a result, claim 1 is not anticipated by *Burton*, and the rejection should be withdrawn.

b. Claims 2-4

Because independent claim 1 is allowable over the cited art of record, dependent claims 2-4 (which depend from independent claim 1, and the remaining dependent claims) are allowable as a matter of law for at least the reason that dependent claims 2-4 contain all the features of independent claim 1. For at least this reason, the rejections of claims 2-4 should be withdrawn.

Additionally and notwithstanding the foregoing reasons for allowability of claims 2-4, these claims recite further features and/or combinations of features (as is apparent by examination of the claim itself) that are patentably distinct from the cited art of record. Hence, there are other reasons why these dependent claims are allowable.

c. Claim 16

Per claim 16, Applicant claims:

A remote object invocation method for invoking a method of a remote object; the method comprising the steps of ***introspecting the remote object to produce introspection data associated with the method, the introspection data including an interface description for the remote object; interpretatively processing the introspection data to establish a proxy object bearing an associated proxy method for the remote object method; invoking, in response to an action of client software, the proxy object method***; conveying invocation data associated with the proxy object method to the remote object; invoking, in response to the invocation data, the method of the remote object; and returning invocation result data to the client software via the object proxy.

(Emphasis added).

Applicant respectfully submits that independent claim 16 is allowable for at least the reason that *Burton* does not disclose, teach, or suggest at least “introspecting the remote object to produce introspection data associated with the method, the introspection data including an interface description for the remote

object; interpretatively processing the introspection data to establish a proxy object bearing an associated proxy method for the remote object method; [and] invoking, in response to an action of client software, the proxy object method,” as recited and emphasized above in claim 16.

Rather, *Burton* describes a system whereby a remote object and a proxy object are instantiated from a preexisting class. *Burton* describes that a client has a preexisting class (e.g., before runtime) for every object which it will ever instantiate. For example, *Burton* states that the “schema further includes proxy object subclasses 54a, 54b, and 54c that implement the proxy class for each of the remote classes 52a, 52b, and 52c. Accordingly, *Burton* fails to teach or suggest “introspecting the remote object to produce introspection data associated with the method, the introspection data including an interface description for the remote object; interpretatively processing the introspection data to establish a proxy object bearing an associated proxy method for the remote object method; [and] invoking, in response to an action of client software, the proxy object method,” as recited and emphasized above in claim 16. (Emphasis added).

Thus, *Burton* fails to teach or suggest all of the features of claim 16. As a result, claim 16 is not anticipated by *Burton*, and the rejection should be withdrawn.

d. Claim 17

Per claim 17, Applicant claims:

A method of invoking, from a first computer, a remote object located on a second computer, the method comprising the steps of ***introspecting, at the second computer, the remote object to identify at least one of a method, property and event thereof to produce introspection data that describes any such identified methods, properties and events; transmitting the introspection data, via a first transport mechanism, to the first computer; creating, at the first computer, a proxy object from the introspection data; invoking, at the first computer, a method of the proxy object; transmitting, from the first computer to the second computer, remote method invocation data, via a second transport mechanism, the remote method invocation data comprising at least an indication of the remote object and the method, property and event thereof to be invoked;*** receiving, at the second computer, via the second transport mechanism, the remote method invocation data, extracting, at the second computer, the remote method invocation data from the second transport data structure; invoking, at the second computer, the method, property or

event of the remote object identified by the remote object invocation data; transmitting, from the second computer to the first computer, via a third transport mechanism, a return object or data representing the results of the invocation of the method, property or event of the remote object; and extracting, at the first computer, the return object or data from the third transport mechanism; wherein the step of creating, at the first computer, the proxy object from the introspection data comprises the step of interpretatively parsing the introspection data and interpretatively constructing the proxy object.

(Emphasis added).

Applicant respectfully submits that independent claim 17 is allowable for at least the reason that *Burton* does not disclose, teach, or suggest at least “introspecting, at the second computer, the remote object to identify at least one of a method, property and event thereof to produce introspection data that describes any such identified methods, properties and events; transmitting the introspection data, via a first transport mechanism, to the first computer; creating, at the first computer, a proxy object from the introspection data; invoking, at the first computer, a method of the proxy object; [and] transmitting, from the first computer to the second computer, remote method invocation data, via a second transport mechanism, the remote method invocation data comprising at least an indication of the remote object and the method, property and event thereof to be invoked,” as recited and emphasized above in claim 17.

Rather, *Burton* describes a system whereby a remote object and a proxy object are instantiated from a preexisting class. *Burton* describes that a client has a preexisting class (e.g., before runtime) for every object which it will ever instantiate. For example, *Burton* states that the “schema further includes proxy object subclasses 54a, 54b, and 54c that implement the proxy class for each of the remote classes 52a, 52b, and 52c. Accordingly, *Burton* fails to teach or suggest “introspecting, at the second computer, the remote object to identify at least one of a method, property and event thereof to produce introspection data that describes any such identified methods, properties and events; transmitting the introspection data, via a first transport mechanism, to the first computer; creating, at the first computer, a proxy object from the introspection data; invoking, at the first computer, a method of the proxy object; [and] transmitting, from the first computer to the second computer, remote method invocation data, via a second transport mechanism, the remote method invocation data comprising at least an indication of the remote object

and the method, property and event thereof to be invoked,” as recited and emphasized above in claim 17. (Emphasis added).

Thus, *Burton* fails to teach or suggest these and other features of claim 17. As a result, claim 17 is not anticipated by *Burton*, and the rejection should be withdrawn.

e. **Claim 19**

Per claim 19, Applicant claims:

A data processing method to invoke a method of a remote object; the method comprising the steps of ***producing remote object data associated with the remote object to discover an object interface dynamically; and interpretatively establishing a proxy object using the remote object data at runtime of client software invoking an associated proxy object method, the proxy object bearing the associated proxy method corresponding to the method of the remote object.***

(Emphasis added).

Applicant respectfully submits that independent claim 19 is allowable for at least the reason that *Burton* does not disclose, teach, or suggest at least “producing remote object data associated with the remote object to discover an object interface dynamically; and interpretatively establishing a proxy object using the remote object data at runtime of client software invoking an associated proxy object method, the proxy object bearing the associated proxy method corresponding to the method of the remote object,” as recited and emphasized above in claim 19.

Rather, *Burton* describes a system whereby a remote object and a proxy object are instantiated from a preexisting class. *Burton* describes that a client has a preexisting class (e.g., before runtime) for every object which it will ever instantiate. For example, *Burton* states that the “schema further includes proxy object subclasses 54a, 54b, and 54c that implement the proxy class for each of the remote classes 52a, 52b, and 52c. Accordingly, *Burton* fails to teach or suggest “producing remote object data associated with the remote object to discover an object interface dynamically; and interpretatively establishing a proxy object using the remote object data at runtime of client software invoking an associated proxy object method, the proxy object bearing the associated proxy method corresponding to the method of the remote object,” as recited in claim 19.

Thus, *Burton* fails to teach or suggest all of the features of claim 19. As a result, claim 19 is not anticipated by *Burton*, and the rejection should be withdrawn.

f. Claims 20-21

Because independent claim 19 is allowable over the cited art of record, dependent claims 20-21 (which depend from independent claim 19) are allowable as a matter of law for at least the reason that dependent claims 20-21 contain all the features of independent claim 19. For at least this reason, the rejections of claims 20-21 should be withdrawn.

Additionally and notwithstanding the foregoing reasons for allowability of claims 20-21, these claims recite further features and/or combinations of features (as is apparent by examination of the claim itself) that are patentably distinct from the cited art of record. Hence, there are other reasons why these dependent claims are allowable.

g. Claim 23

Per claim 23, Applicant claims:

A remote object invocation system to invoke a method of a remote object; the system comprising an inspector to ***produce remote object data associated with the remote object, the remote object data including an interface description for the remote object; an interpreter to interpretatively establishing a proxy object using the remote object data at runtime of client software; the proxy object bearing an associated proxy method corresponding to the method of remote object; a local invocation means to invoke, in response to an action of the client software, the proxy object method; and a carrier to convey invocation data associated with the invocation of the proxy method to the remote object.***

(Emphasis added).

Applicant respectfully submits that independent claim 23 is allowable for at least the reason that *Burton* does not disclose, teach, or suggest at least to “produce remote object data associated with the remote object, the remote object data including an interface description for the remote object; an interpreter to interpretatively establishing a proxy object using the remote object data; the proxy object bearing an associated proxy method corresponding to the method of remote

object; a local invocation means to invoke, in response to an action of client software, the proxy object method; and a carrier to convey invocation data associated with the invocation of the proxy method to the remote object,” as recited and emphasized above in claim 23.

Rather, *Burton* describes a system whereby a remote object and a proxy object are instantiated from a preexisting class. *Burton* describes that a client has a preexisting class (e.g., before runtime) for every object which it will ever instantiate. For example, *Burton* states that the “schema further includes proxy object subclasses 54a, 54b, and 54c that implement the proxy class for each of the remote classes 52a, 52b, and 52c. Accordingly, *Burton* fails to teach or suggest to “produce remote object data associated with the remote object, the remote object data including an interface description for the remote object; an interpreter to interpretatively establishing a proxy object using the remote object data at runtime of client software; the proxy object bearing an associated proxy method corresponding to the method of remote object; a local invocation means to invoke, in response to an action of the client software, the proxy object method; and a carrier to convey invocation data associated with the invocation of the proxy method to the remote object,” as recited and emphasized above in claim 23.

Thus, *Burton* fails to teach or suggest all of the features of claim 23. As a result, claim 23 is not anticipated by *Burton*, and the rejection should be withdrawn.

h. Claim 24

Because independent claim 23 is allowable over the cited art of record, dependent claim 24 (which depends from independent claim 23) is allowable as a matter of law for at least the reason that dependent claim 24 contain all the features of independent claim 23. For at least this reason, the rejection of claim 24 should be withdrawn.

i. **Claim 25**

Per claim 25, Applicant claims:

A remote object server hosting a remote object; the server comprising ***means to generate introspection data associated with the remote object, the introspection data including an interface description for the remote object and means to output the introspection data for use by a client in interpretatively creating a proxy object using the introspection data.***

(Emphasis added).

Applicant respectfully submits that independent claim 25 is allowable for at least the reason that *Burton* does not disclose, teach, or suggest at least “means to generate introspection data associated with the remote object, the introspection data including an interface description for the remote object and means to output the introspection data for use by a client in interpretatively creating a proxy object using the introspection data,” as recited and emphasized above in claim 25.

Rather, *Burton* describes a system whereby a remote object and a proxy object are instantiated from a preexisting class. *Burton* describes that a client has a preexisting class (e.g., before runtime) for every object which it will ever instantiate. For example, *Burton* states that the “schema further includes proxy object subclasses 54a, 54b, and 54c that implement the proxy class for each of the remote classes 52a, 52b, and 52c. Accordingly, *Burton* fails to teach or suggest “means to generate introspection data associated with the remote object, the introspection data including an interface description for the remote object and means to output the introspection data for use by a client in interpretatively creating a proxy object using the introspection data,” as recited and emphasized above in claim 25.

Thus, *Burton* fails to teach or suggest all of the features of claim 25. As a result, claim 25 is not anticipated by *Burton*, and the rejection should be withdrawn.

j. **Claim 26**

Per claim 26, Applicant claims:

A client comprising ***a receiver for receiving introspection data associated with a remote object, the introspection data including an interface description for the remote object, and an interpreter for interpretatively creating a proxy object, using the introspection data, for invocation by an application executable at the client.***

(Emphasis added).

Applicant respectfully submits that independent claim 26 is allowable for at least the reason that *Burton* does not disclose, teach, or suggest at least “a receiver for receiving introspection data associated with a remote object, the introspection data including an interface description for the remote object, and an interpreter for interpretatively creating a proxy object, using the introspection data, for invocation by an application executable at the client,” as recited and emphasized above in claim 26.

Rather, *Burton* describes a system whereby a remote object and a proxy object are instantiated from a preexisting class. *Burton* describes that a client has a preexisting class (e.g., before runtime) for every object which it will ever instantiate. For example, *Burton* states that the “schema further includes proxy object subclasses 54a, 54b, and 54c that implement the proxy class for each of the remote classes 52a, 52b, and 52c. Accordingly, *Burton* fails to teach or suggest “a receiver for receiving introspection data associated with a remote object, the introspection data including an interface description for the remote object, and an interpreter for interpretatively creating a proxy object, using the introspection data, for invocation by an application executable at the client,” as recited in claim 26.

Thus, *Burton* fails to teach or suggest all of the features of claim 26. As a result, claim 26 is not anticipated by *Burton*, and the rejection should be withdrawn.

4. Response To Rejections of Claims Under 35 U.S.C. § 103

In the Office Action, claim 18 stands rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over *Burton* in view of *Haynie* (U.S. Patent Application Publication No. 2006/0036448).

Because independent claim 1 is allowable over the cited art of record, dependent claim 18 (which depends from independent claim 1) is allowable as a matter of law for at least the reason that dependent claim 18 contains all the features of independent claim 1 and *Haynie* fails to cure the deficiencies of the *Burton* reference. For example, *Haynie* is directed to web services and not object components, operations, or methods. For at least this reason, the rejection of claim 18 should be withdrawn.

CONCLUSION

For at least the reasons set forth above, Applicant respectfully submits that all objections and/or rejections have been traversed, rendered moot, and/or accommodated, and that the pending claims are in condition for allowance. Favorable reconsideration and allowance of the present application and all pending claims are hereby courteously requested. If, in the opinion of the Examiner, a telephonic conference would expedite the examination of this matter, the Examiner is invited to call the undersigned agent at (770) 933-9500.

Respectfully submitted,



Charles W. Griggers, Reg. No. 47,283